

A Game-Theoretic Computational Interpretation of Proofs in Classical Analysis

Paulo Oliva and Thomas Powell

Abstract

It is shown in [5, 7] that a functional interpretation of proofs in mathematical analysis can be given by the product of selection functions, a mode of recursion that has an intuitive reading in terms of the computation of optimal strategies in sequential games. We argue that this result has genuine practical value by interpreting some well-known theorems of mathematics and demonstrating that the product gives these theorems a natural computational interpretation that can be clearly understood in game theoretic terms.

1 Introduction

Over the last century, mathematicians and computer scientists have become increasingly interested in understanding the computational content of mathematical proofs.

A central feature of modern mathematics is the use of non-constructive methods that allow us to reason about infinitary objects without providing any computational justification. In the 1920's Hilbert's program broadly addressed the task of understanding non-constructive mathematics in computational terms, which led to the development of important proof-theoretic techniques such as cut-elimination, the ε -method, and proof interpretations. These were used to obtain significant foundational results such as relative consistency proofs for arithmetic and analysis.

In recent decades these metamathematical devices whose roots lie in foundational problems have been employed more directly towards the extraction of programs from non-constructive proofs. This shift of emphasis has its origins in the fundamental work of Kreisel on the 'unwinding' of proofs [16, 17], and has now become the focus of a considerable amount of research in logic and computer science.

Among the most effective tools for extracting constructive information from proofs are the *proof interpretations*, which include Freidman's A-translation [9] and Gödel's dialectica interpretation [11]. The latter in particular is central to the highly successful *proof mining* program (see Kohlenbach [14]), in which the analysis of proofs using its monotone variant has led to the development of quite general meta-theorems guaranteeing the extraction of effective uniform bounds from theorems in analysis.

While proof interpretations have been widely applied in logic and computer science, the *qualitative (computational) behaviour* of their output has received

relatively little attention. Indeed, the operational semantics of programs extracted from even relatively simple classical proofs are often very difficult to understand. This is mainly due to two factors:

1. **Higher-order computation.** Even when computing a witness of type \mathbb{N} the constructions involved will work on much higher types, usually types 1 and 2. Gödel's primitive recursor itself even for the lowest type is already an object of type 2.
2. **Syntactic nature of proof interpretations and translations.** Extracted programs tend to be hidden beneath a complex layer of syntax that generally accompanies formal translations on proofs.

Moreover, relatively little work has gone into addressing these issues because more often than not proof interpretations are a means to an end – be it a consistency proof or the extraction of a uniform bound – and a qualitative understanding of their output is simply irrelevant.

Nevertheless, the idea of stripping functional interpretations of their syntax and appreciating how they work from a *mathematical* perspective is an interesting one. It has been observed by Gaspar and Kohlenbach [10, 14] that the kind of logical manipulations carried out by the dialectica interpretation is closely related to the so-called *correspondence principle* between ‘soft’ and ‘hard’ analysis discussed by T. Tao in [23, 24]. In this sense one could potentially view functional interpretations as devices that transform classical proofs into constructive proofs of a ‘finitized’ form of the original theorem, although actually translating their output into what a mathematician would consider a proof seems far from straightforward.

In recent work [5, 7, 19] the authors and M. Escardó have sought to better understand programs extracted by the dialectica interpretation. Here it is shown that the dialectica interpretation of the key combination of classical logic and countable choice can be realized by the *product of selection functions* (as opposed to the usual bar recursion of Spector [22]), an intuitive mode of computation that can be understood as computing optimal strategies in a class of sequential games.

Consider, for instance, an $\exists\forall$ -theorem $\exists x^X \forall y^Y A(x, y)$, with $A(x, y)$ a decidable predicate, and x and y having types X and Y respectively. We shall think of X as a set of available *moves*, and Y as the set of possible *outcomes* of a game. The predicate $A(x, y)$ is then understood as prescribing what are the good outcomes y given any particular move x . The theorem then says that there exists a single move for which all possible outcomes are considered good. Now, if the theorem has been proven classically, such a move will be shown to exist but it might not be effectively computable. What we should do then is to consider the ‘constructive’ equivalent of the theorem via the negative translation, namely $\neg \neg \exists x^X \forall y^Y A(x, y)$. This can also be put in the form $\exists\forall$, and in fact that is precisely what the dialectica interpretation does. In this case we would obtain the (classically) equivalent theorem

$$\exists \varepsilon^{(X \rightarrow Y) \rightarrow X} \forall p^{X \rightarrow Y} A(\varepsilon p, p(\varepsilon p)).$$

Although $x: X$ might not be effectively computable, it turns out that the *selection function* $\varepsilon: (X \rightarrow Y) \rightarrow X$ is. Moreover, we can extend our game-theoretic

reading and view $p: X \rightarrow Y$ as a mapping from moves to outcomes. What the selection function ε does is to pick, for any given such mapping p , a move $x = \varepsilon p$ whose corresponding outcome according to p , namely $y = px$ is a good outcome for x .

Now suppose we are given a countable family of $\forall\exists$ -predicates $\exists x\forall y A_n(x, y)$ interpreted by a sequence of selection functions (ε_n) . By classical countable choice there exists a sequence $f: \mathbb{N} \rightarrow X$ satisfying $\forall n, y A_n(fn, y)$. The dialectica interpretation of $\neg\exists f\forall n, y A_n(fn, y)$ states that for any given functions $q: X^{\mathbb{N}} \rightarrow Y$ and $\omega: X^{\mathbb{N}} \rightarrow \mathbb{N}$ there exists a functional f such that

$$\forall n \leq \omega f A_n(fn, qf).$$

Therefore, thinking of each $A_n(x, y)$ as prescribing the “good” pairs of move-outcome for round n , the task above corresponds in finding a sequence of moves f which leads to an outcome $y = qf$ that is considered good at all rounds up to point ωf . We will see that the product of the selection functions (ε_n) calculates such f , and this construction can be viewed as the calculation of an optimal strategy in a sequential game whose “goal” at round n – given by the selection function ε_n – is to pick a move with a good outcome for the predicate A_n .

The aim of this article is to demonstrate that, *in practise*, program extraction using the product of selection of functions and its game-theoretic semantics leads to a much better appreciation of the constructive content of proofs in analysis. We illustrate this using a number of well known classical theorems, many of which have been extensively analysed by proof theorists. In particular we include a detailed analysis of a proof of the Bolzano-Weierstrass theorem, the interpretation of which is by no means trivial, but from which we are nevertheless able to extract a program that can be given a clear description in the language of sequential games.

In the course of the paper our aim is to portray the dialectica interpretation as an intelligent translation whose output can be read and understood in mathematical terms. As such we endeavour to phrase the higher type functionals that arise from the interpretation using a more informal vocabulary. This approach owes a lot to the aforementioned work by Gaspar and Kohlenbach, and it is hoped that our work will complement theirs in forming another small step towards understanding the mathematical significance of proof interpretations.

1.1 Preliminaries

We work in the language of Peano arithmetic in all finite types \mathbf{PA}^ω . The finite types \mathbf{T} contain a basic type \mathbb{N} and whenever $X, Y \in \mathbf{T}$ then $X \rightarrow Y \in \mathbf{T}$, i.e.

$$\mathbf{T} = \{\mathbb{N}, \mathbb{N} \rightarrow \mathbb{N}, (\mathbb{N} \rightarrow \mathbb{N}) \rightarrow \mathbb{N}, \dots\}.$$

Closely related to \mathbf{PA}^ω is Gödel’s system \mathbf{T} of primitive recursive functionals of finite type. This quantifier-free calculus is essentially primitive recursive arithmetic PRA with the schema of recursion extended to all types $X \in \mathbf{T}$, i.e.

$$R_n(h)(g) \stackrel{X}{=} \begin{cases} g & \text{if } n = 0 \\ h_n(R_{n-1}(h)(g)) & \text{if } n > 0. \end{cases} \quad (1)$$

For full details of these theories the reader is referred to [1]. We make informal use of types like \mathbb{B} , \mathbb{Q} , finite sequences types X^* etc. as elements of these types

can be encoded as elements of a suitable type in \mathbf{T} .

Notation. We make use of the following abbreviations:

0_X is the obvious canonical zero of type X .

$s * t$ is the concatenation of sequences s and t .

$s \preceq t$ for “ s is a prefix of t ”.

$\widehat{s} \equiv s * \mathbf{0}^{X^\mathbb{N}}$ the canonical infinite extension of a finite sequence s .

$[\alpha](n) \equiv \langle \alpha 0, \dots, \alpha(n-1) \rangle$ is the initial segment of α of length n .

$\mu n \leq N . P(n)$ is the bounded search operator that returns the least $n \leq N$ satisfying the decidable predicate $P(n)$ if one exists, or N otherwise.

1.2 The dialectica interpretation

We assume that the reader is familiar with Gödel’s dialectica interpretation (details of which are covered in full in [1, 14]), although we recall below a few basic facts to familiarise the reader with our notation and terminology.

The dialectica interpretation maps formulas A of some specified theory \mathcal{S} to a decidable binary relation $|A|_y^x$ definable in a specified quantifier-free system of functionals \mathbf{F} . The canonical instance of this mapping is when \mathcal{S} is Heyting arithmetic in all finite types and \mathbf{F} is Gödel’s system \mathbf{T} .

In $|A|_y^x$ we have that x and y stand for (possibly empty) tuples of objects of finite type. We think of x as the *witnessing* variables and y as the *challenge* variables. The intuition is that A is logically equivalent to $\exists x \forall y |A|_y^x$. The translation is formally defined as follows:

Definition 1 (Gödel’s dialectica interpretation). *For atomic formulas P we set $|P| := P$, with x and y both empty tuples. Assuming that we have already defined $|A|_y^x$ and $|B|_v^u$, we define*

$$\begin{aligned} |A \wedge B|_{y,w}^{x,v} &:= |A|_y^x \wedge |B|_w^v \\ |A \vee B|_{y,w}^{x,v,b} &:= (b = 0 \wedge |A|_y^x) \vee (b = 1 \wedge |B|_w^v) \\ |A \rightarrow B|_{x,w}^{f,g} &:= |A|_{gxw}^x \rightarrow |B|_w^{fx} \\ |\forall z A(z)|_{y,z}^f &:= |A(z)|_y^{fz} \\ |\exists z A(z)|_y^{x,z} &:= |A(z)|_y^x. \end{aligned}$$

We say that \mathcal{S} is (dialectica) interpreted in \mathbf{F} if whenever $\mathcal{S} \vdash A$ we can construct some $t \in \mathbf{F}$ such that $\mathbf{F} \vdash |A|_y^t$.

In order to interpret *classical* theories, the dialectica interpretation is typically composed with a negative translation¹ N to form the so-called ND interpretation. In the remainder of the paper, by *functional interpretation* we specifically mean the ND interpretation. A classical theory \mathcal{T} has a functional interpretation in \mathbf{F} if whenever $\mathcal{T} \vdash A$ we can construct some $t \in \mathbf{F}$ satisfying $\forall y |A^N|_y^t$.

¹As in [14] we adopt Kuroda’s variant of the negative translation.

In his original paper on the dialectica interpretation, Gödel proved that Peano arithmetic has a functional interpretation in the primitive recursive functionals of finite type T . Later, Spector extended Gödel's result to classical analysis by realizing the dialectica interpretation of the negative translation of the axiom of countable choice AC_0 with a novel, but rather abstruse form of recursion called *bar recursion* SBR .

Theorem 2. *The following are well-known:*

- (a) PA^ω has a functional interpretation in T (Gödel [11]).
- (b) $PA^\omega + AC_0$ has a functional interpretation in $T + SBR$ (Spector [22]).

The main purpose of this paper is to show that these soundness theorems can be reformulated in terms of the product of selection functions, and that this reformulation is better suited towards understanding the behaviour of programs extracted by the dialectica interpretation.

1.3 Outline of article

We begin in Section 2 by introducing the product of selection functions and showing that it can be characterised as an operation that computes optimal strategies in sequential games.

In the main part of the paper we then discuss how the language of selection functions is well suited to capturing the way in which the dialectica interpretation works, and in particular the product of selection functions directly interprets countable choice.

We then present a short case study (Section 4) in which we extract a program from a proof of the Bolzano-Weierstrass theorem via the product of selection functions and demonstrate that our program has a clear game-theoretic semantics.

We conclude by briefly discussing some of the problems we face in gaining a more intuitive understanding of functional interpretations, and outline some potential directions for further research.

2 Selection Functions and Sequential Games

This section constitutes a brief overview of work that is presented in full elsewhere, e.g. the reader is referred to the original paper [6] or a recent survey [7] for a more detailed treatment.

A *selection function* is defined to be any element of type $(X \rightarrow R) \rightarrow X$ (as in [6] we abbreviate this type to J_RX). Closely related to a selection function $\varepsilon: J_RX$ is its corresponding *quantifier* $\bar{\varepsilon}: (X \rightarrow R) \rightarrow R$ defined by $\bar{\varepsilon}p := p(\varepsilon p)$. The intuition is to view ε as a selector that given a function $p: X \rightarrow R$ picks a particular element of εp of X that *attains* its quantifier $\bar{\varepsilon}$, as the following examples illustrate.

Example 3. (a) The canonical example of a selection function and its associated quantifier is when R forms a set of truth values e.g. $R = \mathbb{B}$. Hilbert's

epsilon term of type X , $\varepsilon_X: J_{\mathbb{B}}X$ is a selection function whose corresponding quantifier is just the usual existential quantifier \exists_X for predicates over type X , since by definition we have

$$\exists x^X p(x) \Leftrightarrow p(\varepsilon_X p).$$

- (b) By the mean value theorem there exists a selection function $\varepsilon: J_{[0,1]}\mathbb{R}$ such that for any continuous function $p: [0, 1] \rightarrow \mathbb{R}$ we have

$$p(\varepsilon p) = \int_0^1 p(x)dx.$$

Its corresponding quantifier is the operator \int_0^1 .

- (c) Assume we are given a position in a game where we have to pick a move in X . A *strategy* for that position can be defined by a selection function $\varepsilon: J_RX$ determining an optimal move for each given mapping $p: X \rightarrow R$ of possible moves $x \in X$ to corresponding outcomes $p(x) \in R$.

The theory of selection functions and quantifiers forms the basis of [5, 6, 7]. One of the main achievements of these papers has been to define a product operation on selection functions (along with a corresponding operation on quantifiers which we do not discuss further here). They demonstrate that the product of selection functions is an extremely versatile construction that appears naturally in several different areas of mathematics and computer science, such as fixed point theory (Bekić's lemma), algorithms (backtracking), game theory (backward induction) and, as we also discuss in Section 3, proof theory.

In the remainder of the section we define (following [6]) the product of selection functions, and explain how this procedure can be best understood via the computation of optimal strategies in a certain class of sequential games.

Definition 4 (Binary product of selection functions [6]). *Given a selection function $\varepsilon: J_RX$ and family of selection functions $\delta_x: J_RY$ and a predicate $q: X \times Y \rightarrow R$, let*

$$\begin{aligned} A[x^X] &:= \delta(x, \lambda y. q(x, y)) \\ a &:= \varepsilon(\lambda x. q(x, A[x])). \end{aligned}$$

The binary product $\varepsilon \otimes \delta$ is another selection function, of type $J_R(X \times Y)$, defined by

$$(\varepsilon \otimes \delta)(q) \stackrel{X \times Y}{:=} \langle a, A[a] \rangle.$$

If δ is independent of x we can this the simple product of selection functions. The general case is then also called the dependent product of selection functions.

The binary product constructs a composite selection function on the type $X \times Y$ in the obvious way:

Example 5. Continuing from Example 3 we have:

- (a) It is easy to show that the product of ε operators $\varepsilon_X \otimes \varepsilon_Y$ is an epsilon operator of type $X \times Y$ in the sense that

$$\exists x^X \exists y^Y q(x, y) \Leftrightarrow q((\varepsilon_X \otimes \varepsilon_Y)(q)).$$

(b) Given a continuous function $q: [0, 1]^2 \rightarrow \mathbb{R}$ we have

$$q((\varepsilon \otimes \varepsilon)(q)) = \int_0^1 \int_0^1 q(x, y) \, dx dy.$$

(c) Given strategies $\varepsilon_0, \varepsilon_1$ for each round in a two round sequential game with outcome function $q: X_0 \times X_1 \rightarrow R$, then $(\varepsilon_0 \otimes \varepsilon_1)(q)$ forms an *strategy* for the game which is “compatible” with the local strategies ε_0 and ε_1 . This key instance of the product is discussed in more detail below.

As described in [6], we can iterate the binary product of selection functions a finite or an unbounded number of times, where the length of the iteration is dependent on the output of the product in the following sense.

Definition 6 (Iterated product of selection functions [6]). *Suppose we are given a family of selection functions $(\varepsilon_s: J_R X)$. The explicitly controlled unbounded product of the selection functions ε_s is defined by the recursion schema*

$$\text{EPS}_s^\omega(\varepsilon)(q) \stackrel{X^\mathbb{N}}{=} \begin{cases} \mathbf{0} & \text{if } \omega(\widehat{s}) < |s| \\ (\varepsilon_s \otimes \lambda x. \text{EPS}_{s*x}^\omega(\varepsilon))(q) & \text{otherwise} \end{cases} \quad (2)$$

where $s: X^*$, $q: X^\mathbb{N} \rightarrow R$ and $\omega: X^\mathbb{N} \rightarrow \mathbb{N}$.

The functional ω acts as a control, terminating the procedure once it has produced a sequence s satisfying $\omega(\widehat{s}) < |s|$. The unbounded product is total in any model of bar recursion, which in particular must admit *Spector’s condition*:

$$\forall \omega^{X^\mathbb{N} \rightarrow \mathbb{N}}, \alpha^{X^\mathbb{N}} \exists n \left(\omega(\widehat{[\alpha](n)}) < n \right).$$

These include the models of continuous functionals and the majorizable functionals. On the other hand, when ω is a constant function, say $\omega\alpha = n$, this corresponds to a finite iteration of the binary product and this restricted instance of the product is definable in the primitive recursive functionals and therefore exists in any model of system T.

By unwinding the definition of the binary product in (2) we obtain an equivalent equation

$$\text{EPS}_s^\omega(\varepsilon)(q) \stackrel{X^\mathbb{N}}{=} \begin{cases} \mathbf{0} & \text{if } \omega(\widehat{s}) < |s| \\ a_s * \text{EPS}_{s*a_s}^\omega(\varepsilon)(q_{a_s}) & \text{otherwise} \end{cases} \quad (3)$$

where $a_s = \varepsilon_s(\lambda x. \overline{\text{EPS}_{s*x}^\omega(\varepsilon)}(q_x))$ and $q_a(\alpha) = q(a * \alpha)$.

For fixed ω, ε and q one can think of $\text{EPS}_s^\omega(\varepsilon)(q)$ as computing an infinite extension to any given finite sequence s . The key property of EPS is that the infinite extension of an initial segment $[\alpha](n)$ of a previous infinite extension α is identical to the original infinite extension. Formally:

Lemma 7 (cf. [22], lemma 1). *Let $\alpha = \text{EPS}_s^\omega(\varepsilon)(q)$. For all n ,*

$$\alpha = [\alpha](n) * \text{EPS}_{s*[\alpha](n)}^\omega(\varepsilon)(q_{[\alpha](n)}). \quad (4)$$

Proof. By induction on n . If $n = 0$ this follows by the definition of α . Assume (4) holds for n , we wish to show it also holds for $n + 1$. Consider two cases.

(a) If $\omega(s * [\alpha](n) * \mathbf{0}) < n$ then $\text{EPS}_{s*[\alpha](n)}^\omega(\varepsilon)(q_{[\alpha](n)}) = \mathbf{0}^{X^\mathbb{N}}$. By induction hypothesis $\alpha = [\alpha](n) * \mathbf{0}$, so that $\alpha(n) = \mathbf{0}^X$. Therefore $\alpha = [\alpha](n + 1) * \mathbf{0}$, which, by extensionality, implies $\omega(s*[\alpha](n+1)*\mathbf{0}) = \omega(s*[\alpha](n)*\mathbf{0}) < n < n+1$. Hence, $\text{EPS}_{s*[\alpha](n+1)}^\omega(\varepsilon)(q_{[\alpha](n+1)}) = \mathbf{0}^{X^\mathbb{N}}$ so that

$$[\alpha](n + 1) * \text{EPS}_{s*[\alpha](n+1)}^\omega(\varepsilon)(q_{[\alpha](n+1)}) = [\alpha](n + 1) * \mathbf{0} = [\alpha](n) * \mathbf{0} = \alpha.$$

(b) If $\omega(s * [\alpha](n) * \mathbf{0}) \geq n$, then

$$\alpha \stackrel{(\text{IH})}{=} [\alpha](n) * \text{EPS}_{s*[\alpha](n)}^\omega(\varepsilon)(q_{[\alpha](n)}) \stackrel{(3)}{=} [\alpha](n) * c * \text{EPS}_{s*[\alpha](n)*c}^\omega(\varepsilon)(q_{[\alpha](n)*c}),$$

where $c = \varepsilon_{s*[\alpha](n)}(\lambda x. \overline{\text{EPS}_{s*[\alpha](n)*x}^\omega(\varepsilon)}(q_{s*[\alpha](n)*x}))$. Hence, $\alpha(n) = c$. Therefore

$$\alpha = [\alpha](n + 1) * \text{EPS}_{s*[\alpha](n+1)}^\omega(\varepsilon)(q_{[\alpha](n+1)}).$$

□

This lemma is the main building block behind the proof of the following fundamental theorem about EPS.

Theorem 8 (Main theorem on EPS). *Let $q: X^\mathbb{N} \rightarrow R$ and $\omega: X^\mathbb{N} \rightarrow \mathbb{N}$ and $\varepsilon_s: J_R X$ be given. Define*

$$\begin{aligned} \alpha &\stackrel{X^\mathbb{N}}{=} \text{EPS}_{\langle \rangle}^\omega(\varepsilon)(q) \\ p_s(x) &\stackrel{R}{=} \overline{\text{EPS}_{s*x}^\omega(\varepsilon)}(q_{s*x}). \end{aligned}$$

For $n \leq \omega(\alpha)$ we have

$$\begin{aligned} \alpha(n) &\stackrel{X}{=} \varepsilon_{[\alpha](n)}(p_{[\alpha](n)}) \\ q\alpha &\stackrel{R}{=} \overline{\varepsilon_{[\alpha](n)}}(p_{[\alpha](n)}). \end{aligned} \tag{5}$$

Proof. Assume $n \leq \omega(\alpha)$. We argue that $(*)$ $n \leq \omega([\alpha](n) * \mathbf{0})$. Otherwise, assuming $n > \omega([\alpha](n) * \mathbf{0})$ we would have, by Lemma 7, that $\alpha = [\alpha](n) * \mathbf{0}$. And hence, $n > \omega([\alpha](n) * \mathbf{0}) = \omega(\alpha) \geq n$, which is a contradiction. Hence, we have that

$$\begin{aligned} \alpha(n) &\stackrel{\text{L7}}{=} \text{EPS}_{[\alpha](n)}^\omega(\varepsilon)(q_{[\alpha](n)})(0) \\ &\stackrel{(3)+(*)}{=} \varepsilon_{[\alpha](n)}(\lambda x. \overline{\text{EPS}_{[\alpha](n)*x}^\omega(\varepsilon)}(q_{[\alpha](n)*x})) \\ &= \varepsilon_{[\alpha](n)}(p_{[\alpha](n)}), \end{aligned}$$

by the definition of p_s . For the second equality, we have

$$\begin{aligned} q\alpha &\stackrel{\text{L7}}{=} q_{[\alpha](n+1)}(\text{EPS}_{[\alpha](n+1)}^\omega(\varepsilon)(q_{[\alpha](n+1)})) \\ &= p_{[\alpha](n)}(\alpha(n)) \\ &= \overline{\varepsilon_{[\alpha](n)}}(p_{[\alpha](n)}), \end{aligned}$$

where the last equality uses that $\alpha(n) = \varepsilon_{[\alpha](n)}(p_{[\alpha](n)})$ is already shown. □

Theorem 8 characterises the product of selection functions as computing a sequence α that forms a kind of sequential equilibrium between the selection functions – expressed by the equations (5) – up to a point $\omega\alpha$ parametrised by α itself. The significance of the product is that such equilibria appear naturally in a variety of contexts. In the following we outline perhaps the most illuminating of these contexts, namely the theory of sequential games.

2.1 Sequential games and optimal strategies

One of the most remarkable property of EPS is that it computes optimal strategies in a certain class of sequential games. The reader is encouraged to consult [7] in conjunction with the relatively concise discussion here.

As in this article we only consider games (in the sense of [7]) where the quantifiers are attainable, we shall incorporate this restriction in the definition of the game itself.

Definition 9 (Sequential games of unbounded length, [7]). *The type of a game is given by a pair (X, R) where*

- X is the set of possible moves at each round.
- R is the set of possible outcomes of the game.

A finite sequence $s: X^$ shall be thought of as a position in the game determined by the first $|s|$ moves. An infinite sequence $\alpha: X^{\mathbb{N}}$ is called a play of the game. An unbounded sequential game of type (X, R) is a triple (ε, q, ω) where*

- $\varepsilon_s: J_R X$ determines the optimal move at position s .
- $q: X^{\mathbb{N}} \rightarrow R$ determines, given a play $\alpha: X^{\mathbb{N}}$, the outcome of the game.
- $\omega: X^{\mathbb{N}} \rightarrow \mathbb{N}$ determines the relevant part of a play.

The functions q is called the outcome function, whereas ω is called the control function. Given a play α , all moves $\alpha(i)$ for $i \leq \omega\alpha$ are relevant moves. In general, a position s is called relevant if $|s| \leq \omega\hat{s}$, i.e. if in a canonical extension of the current position s the current move is considered a relevant move.

We shall only consider infinite plays which are obtained by some canonical extension of a finite play s . Therefore, we think of these as finite games of unbounded length.

The intuition behind Definition 9 is as follows. We think of the selection functions ε_s as specifying at position s what an optimal move at that point would be if we knew the final outcome corresponding to each of the candidate moves. The selection function takes this mapping $X \rightarrow R$ of moves to outcomes and tells us what an optimal move would be in that particular case.

A *strategy* in such game is simply a function $\text{next}: X^* \rightarrow X$ which determines for each position s what the next move $\text{next}(s)$ should be. To follow a strategy from position s means to play all following moves according to the strategy, i.e. we obtain a sequence of moves $\alpha(0), \alpha(1), \dots$ as

$$\alpha(i) = \text{next}(s * [\alpha](i-1)).$$

We call this the *strategic extension* of s . The strategic extension of the empty play is called the *strategic play*.

Definition 10 (Optimal strategies). *A strategy is said to be optimal² if the move played at each relevant position s is the one recommended by the selection function ε_s , i.e.*

$$\text{next}(s) = \varepsilon_s(\lambda x. q(s * x * \alpha)) \quad (6)$$

where α is the strategic extension of $s * x$.

The main result of [7] is that the product of selection functions computes optimal strategies:

Theorem 11 ([7]). *Given a game (ε, q, ω) , the strategy*

$$\text{next}(s) \stackrel{X}{=} (\text{EPS}_s^\omega(\varepsilon)(q))_0 \quad (7)$$

is optimal, and, moreover,

$$\alpha \stackrel{X^N}{=} \text{EPS}_s^\omega(\varepsilon)(q_s) \quad (8)$$

is the strategic extension of s , i.e. $\alpha(n) = \text{next}(s * [\alpha](n))$.

Proof. We have that

$$\begin{aligned} \alpha(n) &\stackrel{(8)}{=} \text{EPS}_s^\omega(\varepsilon)(q_s)(n) \\ &\stackrel{L7}{=} \text{EPS}_{s*[\alpha](n)}^\omega(\varepsilon)(q_{s*[\alpha](n)})(0) \\ &\stackrel{(7)}{=} \text{next}(s * [\alpha](n)), \end{aligned}$$

which proves the second claim. Hence, assuming s is a relevant position, i.e. $(*) \ \omega(\hat{s}) \geq |s|$ we have

$$\begin{aligned} \text{next}(s) &\stackrel{(7)}{=} (\text{EPS}_s^\omega(\varepsilon)(q_s))_0 \\ &\stackrel{(3)+(*)}{=} \varepsilon_s(\lambda x. \overline{\text{EPS}_{s*x}^\omega(\varepsilon)}(q_{s*x})) \\ &= \varepsilon_s(\lambda x. q(s * x * \beta)) \end{aligned}$$

where $\beta = \text{EPS}_{s*x}^\omega(\varepsilon)(q_{s*x})$, by the second claim just proven above, is the strategic extension of $s * x$. Hence, we have shown (6). \square

Therefore in this sense the main Theorem 8 characterises EPS as a procedure that computes an optimal strategy in the game defined by (ε, q, ω) . We now show that the product also appears naturally in proof theory, with the advantage that it can be related back to the language of sequential games.

3 The dialectica interpretation of classical proofs

We now show how selection functions and their product are intrinsically connected to the functional interpretation of classical proofs. The key observation is that the language of selection functions elegantly captures the way in which

²This is a stronger notion than the one introduced in [7] for the more general case where the quantifiers are not necessarily attainable.

the dialectica interpretation treats double negations in negative-translated formulas. In particular the product directly interprets the double negation shift that arises from the negative translation of the axiom of countable choice.

This means that in many cases the algorithms extracted from classical proofs can be easily phrased in the intuitive language of sequential games. Moreover, though couched in the language of higher type recursive functionals, these games often have a natural informal reading in terms of strategic set-theoretic constructions, making the *mathematical* meaning of the extracted program more perspicuous.

3.1 Interpreting Σ_2 theorems

Suppose are given a Σ_2 theorem $A \equiv \exists x^X \forall y^Y A_0(x, y)$ where A_0 is decidable. The negative translation of A is equivalent to $\neg \neg \exists x \forall y A_0(x, y)$, and therefore its functional interpretation is given by

$$|A^N|_p^\varepsilon = A_0(\varepsilon p, p(\varepsilon p)).$$

In other words, the dialectica interpretation eliminates double negations in front of a Σ_2 formula with a selection function $\varepsilon: J_Y X$. If the predicate $A_0(x, y)$ is thought of as prescribing ‘good’ outcomes y for a particular move x as described in Section 1, then ε implements a strategy that selects a move $x = \varepsilon p$ whose outcome with respect to the mapping p is good.

Thus under the functional interpretation we have the following mapping:

$$\Sigma_2 \text{ Theorems} \mapsto \text{Selection functions.}$$

The elimination of double negations in an arbitrary negated formula is essentially a (albeit complex) modular iteration of this process, suggesting to us that selection functions and modes of recursion based on selection functions lie behind the functional interpretation in a fundamental way.

There are several ways of characterising the selection function ε interpreting A . For Σ_2 theorems Kreisel’s *no counterexample interpretation* coincides with the functional interpretation and in this sense the constructive interpretation of A is a selection function ε that refutes an arbitrary ‘counterexample’ functions p . The following example demonstrates how selection functions are fundamental to the functional interpretation of pure classical logic.

Example 12 (Law of excluded middle). Consider the following simple reformulation of the law of excluded middle for Σ_1 formulas, better known as the *drinkers paradox*:

$$\text{DP} : \quad \exists x^X (\exists y P(y) \rightarrow P(x)). \quad (9)$$

Note that DP is intuitionistically equivalent to the Σ_2 theorem $\exists x \forall y (P(y) \rightarrow P(x))$. We ineffectively justify the principle by defining

$$x := \begin{cases} y & \text{for some } y \text{ satisfying } P(y) \\ 0_X & \text{if no such } y \text{ exists} \end{cases}.$$

On the other hand, we can *effectively* justify the principle with the selection function

$$\varepsilon p := \begin{cases} p(0_X) & \text{if } P(p(0_X)) \\ 0_X & \text{if } \neg P(p(0_X)) \end{cases} \quad (10)$$

that witnesses its functional interpretation:

$$|\text{DP}|_p^\varepsilon = P(p(\varepsilon p)) \rightarrow P(\varepsilon p). \quad (11)$$

The drinkers paradox is essentially the law of excluded middle applied to the Σ_1 -formula $\exists y P(y)$ i.e.

$$\exists b^{\mathbb{B}} (b = 0 \leftrightarrow \exists y P(y))$$

where the boolean b is given by $P(x)$. The mapping $p: X \rightarrow X$ in the functional interpretation of DP can be seen as a *counterexample function* that attempts to witness $\neg \text{DP}$ i.e.

$$\forall x (\neg P(x) \wedge P(px)). \quad (12)$$

The constructive version of the law of excluded middle given by its functional interpretation is the statement that for any n, p there exists an element x refuting (12):

$$P(x) \vee \neg P(px).$$

The selection function ε witnesses this statement.

One can alternatively view the selection function ε interpreting A as an algorithm that produces an arbitrary large *approximation* to the ineffective object x satisfying $\forall y A_0(x, y)$. In fact, when $Y = \mathbb{N}$ the formula A is equivalent to $\exists x^X \forall y \forall i \leq y A_0(x, i)$. Hence, the functional interpretation of A is equivalent to the existence of a selection functions ε satisfying

$$\forall p \forall i \leq p(\varepsilon p) A_0(\varepsilon p, i).$$

We see p as a function that specifies in advance how we want to use A in a particular computation, and ε returns a sufficiently high quality approximation to x . This reading is closer to the notion of a ‘finitization’ of as discussed by Tao in [23], in the sense that we interpret the qualitative statement that there exists some x with the permanent property $\forall y A_0(x, y)$ by the quantitative statement that there exist approximations x with the temporary property $\forall i \leq px A_0(x, i)$ for arbitrary p .

Example 13 (Convergence and metastability). The functional interpretation of Cauchy convergence

$$\forall k > 0 \exists n \forall i, j \geq n (\|x_i - x_j\| \leq 2^{-k})$$

is a sequence of selection functions (ε_k) that satisfy

$$\forall k > 0, p \forall i, j \in [\varepsilon_k p, \varepsilon_k p + p(\varepsilon_k p)] (\|x_i - x_j\| \leq 2^{-k}). \quad (13)$$

In other words, the Cauchy convergence property is equivalent to the existence of a sequence of selection functions ε_k that compute regions of approximate stability, or *metastability*, of size specified by p .

This reformulation of convergence plays a key role in ergodic theory, where one obtains quantitative versions of convergence theorems by extracting explicit bounds on $\varepsilon_k p$ that are highly uniform with respect to (x_n) . A simple example is the so-called ‘finite convergence principle’ discussed in [14, 23], where one can easily show that given k and p a bounded monotone sequence

$$0 \leq x_0 \leq x_1 \leq \dots \leq 1$$

experiences a period of metastability bounded uniformly by $\tilde{p}(2^k + 1)(0)$ for $\tilde{p}(n) : +p(n)$.

A more involved example of the extraction of uniform bounds on the selection functions is, for instance, the quantitative mean ergodic theorem proved by Avigad et al. in [2].

3.2 Interpreting the axiom of choice

Classical predicate logic PL^ω can be extended to encompass most of mathematics through the addition of choice principles. In particular the principle of *finite* choice FC is known to be equivalent to induction and therefore we can define Peano arithmetic (assuming a minimal amount of arithmetic) as $\text{PA}^\omega := \text{PL}^\omega + \text{FC}$, while the further addition of countable choice AC_0 yields a theory sufficient to formalise a large portion of analysis.

Thus a key part of understanding the computational content of classical proofs is to understand the computational interpretation of the axiom of countable choice combined with classical logic.

Let us first consider an instance of AC_0 for Π_1 formulas:

$$\Pi_1\text{-AC}_0 : \forall n \exists x^X \forall y^Y A_n(x, y) \rightarrow \exists f^{\mathbb{N} \rightarrow X} \forall n, y A_n(fn, y),$$

for decidable A_n . Its negative translation is equivalent to

$$\forall n \neg \neg \exists x \forall y A_n(x, y) \rightarrow \neg \neg \exists f \forall n, y A_n(fn, y),$$

and its dialectica interpretation is equivalent (using just Markov's principle, which is admitted by the dialectica interpretation) to the statement

$$\forall \varepsilon, q, \omega \exists f (\forall n, p A_n(\varepsilon_n p, p(\varepsilon_n p)) \rightarrow \forall i \leq \omega f A_i(fi, qf)). \quad (14)$$

This constructive interpretation of AC_0 asks for a selection function $F^\varepsilon : J_{Y \times \mathbb{N}} X^\mathbb{N}$ producing an approximation to the sequence f , given selection functions ε interpreting its premise. Such a selection function can be given by

$$F^\varepsilon(q, \omega) := \text{EPS}_\omega^\omega(\varepsilon)(q).$$

We now prove in detail that the product of selection functions directly realizes the functional interpretation of the axiom of choice.

Theorem 14. *The following hold:*

(a) *The functional interpretation of the schema of finite choice*

$$\text{FC} : \forall n \leq N \exists x^X A_n(x) \rightarrow \exists s^{X^*} \forall n \leq N A_n(s_n)$$

is directly witnessed by the finite simple product of selection functions (i.e. ω a constant function).

(b) *The functional interpretation of the schema of countable choice*

$$\text{AC}_0 : \forall n \exists x^X A_n(x) \rightarrow \exists f^{X^\mathbb{N}} \forall n A_n(fn)$$

is directly witnessed by the (unbounded) simple product of selection functions.

(c) *The functional interpretation of the schema of dependent choice*

$$\text{DC} : \forall s^{X^*} \exists x^X A_s(x) \rightarrow \exists f^{X^\mathbb{N}} \forall n A_{[f](n)}(fn)$$

is directly witnessed by the dependent product of selection functions.

Proof. We prove (c), the other parts are particular cases of this. Since $A_s(x)$ is equivalent to $\exists \tilde{x}^{\tilde{X}} \forall y |A_s(x)|_{\tilde{y}}^N$ it suffices to interpret DC for Σ_2 -formulas

$$\Sigma_2\text{-DC} : \forall s^{X^*} \exists x^X, \tilde{x}^{\tilde{X}} \forall y |A_s(x)|_{\tilde{y}}^N \rightarrow \exists f^{X^\mathbb{N}} \forall n \exists \tilde{x}^{\tilde{X}} \forall y |A_{[f](n)}(fn)|_{\tilde{y}}^N.$$

Moreover, by adding a dummy variable t of type \tilde{X}^* and concatenating the types X, \tilde{X} this follows directly from an instance of $\Pi_1\text{-DC}$ i.e.

$$\Pi_1\text{-DC} : \forall s^{X^*}, t^{\tilde{X}^*} \exists x, \tilde{x} \forall y |A_s(x)|_{\tilde{y}}^N \rightarrow \exists f^{X^\mathbb{N}}, \tilde{f}^{\tilde{X}^\mathbb{N}} \forall n, y |A_{[f](n)}(fn)|_{\tilde{y}}^N.$$

Therefore it suffices to deal with $\Pi_1\text{-DC}$, which in general has a negative translation equivalent to

$$\Pi_1\text{-DC}^N : \forall s^{X^*} \neg \neg \exists x^X \forall y A_s(x, y) \rightarrow \neg \neg \exists f \forall n, y A_{[f](n)}(fn, y).$$

The dialectica interpretation of $\Pi_1\text{-DC}^N$ is equivalent to

$$|\text{DC}^N|_{\varepsilon, \omega, q}^{F, p, s} \equiv A_s(\varepsilon_s p, p(\varepsilon_s p)) \rightarrow A_{[F](\omega F)}(F(\omega F), qF), \quad (15)$$

omitting, for the sake of readability, the parameters ε, ω and q from the functions F, p and s . In fact, these parameters (ε, ω, q) define a sequential game in the sense of Definition 9. Therefore, let

$$\begin{aligned} F &= \text{EPS}_{\langle \rangle}^\omega(\varepsilon)(q) \\ p_s(x) &= \overline{\text{EPS}_{s*x}^\omega(\varepsilon)}(q_{s*x}). \end{aligned}$$

By Theorem 8 we have that F and $p := p_{[F](\omega F)}$ and $s := [F](\omega F)$ are such that $\varepsilon_s p = F(\omega F)$ and $p(\varepsilon_s p) = qF$, and hence, clearly witness $|\text{DC}^N|_{\varepsilon, \omega, q}^{F, p, s}$. An analogous but simpler argument proves (1) and (2), proofs of which can also be found in [8] and [5] respectively. \square

Theorem 14 proves that under the functional interpretation we have a mapping

$$\text{Choice principles} \mapsto \text{Product of selection functions.}$$

At first glance it may seem strange that an operation that computes optimal strategies in sequential games is related to the axiom of choice in this manner. But if we take a closer look, the game theoretic behaviour of (14) becomes clear. The selection functions ε_n which realise the premise of (14) can be seen as a collection of strategies each witnessing the Σ_2 theorems (A_n) . The dialectica interpretation calls for a procedure that takes these pointwise strategies and produces a co-operative selection function F that witnesses $\forall n A_n$. Such a procedure is provided naturally by the product of selection functions.

In the following examples we illustrate how the interpretation of theorems that make direct use of the axiom of choice can be given an intuitive game-theoretic constructive interpretation by the product of selection functions.

Example 15 (Arithmetic comprehension). We first give a realizer for the functional interpretation of arithmetic comprehension for Σ_1^0 formulas, which states that for any Σ_1 predicate φ over \mathbb{N} there exists a set X with

$$\forall n(n \in X \Leftrightarrow \exists y \varphi(n, y)).$$

Computing such X is in general not possible. We can, however, try to compute an “approximation” to X . For instance, we might ask for an \tilde{X} which only works for a finite number of n ’s, or an approximation which only checks the existence of y ’s up to a certain bound (possibly depending on the approximating set \tilde{X}). We call these calibrations of the ‘size’ and ‘depth’ of X , respectively.

Arithmetic comprehension follows from the formal statement

$$\exists f^{\mathbb{N} \rightarrow \mathbb{N}} \forall n(\exists y \varphi(n, y) \rightarrow \exists k < fn \varphi(n, k)),$$

where we define $X := \{n \mid \exists k < fn \varphi(n, k)\}$. Again, we cannot (in general) effectively construct X . Indeed, the above is a direct consequence of countable choice applied the non-constructive statement

$$\forall n \exists x^{\mathbb{N}} (\exists y \varphi(n, y) \rightarrow \exists k < x \varphi(n, k)). \quad (16)$$

But this is just a collection of instances of DP applied to the formulas $P_n(x) := \exists k < x \varphi(n, k)$. Therefore defining the sequence of selection functions (ε_n) by

$$\varepsilon_n p := \begin{cases} 0 & \text{if } \forall k < p(0) \neg \varphi(n, k) \\ p(0) & \text{if } \exists k < p(0) \varphi(n, k) \end{cases}$$

we have

$$\exists k < p(\varepsilon_n p) \varphi(n, k) \rightarrow \exists k < \varepsilon_n p \varphi(n, k)$$

for any n, p , and thus by Theorem 8, for any counterexample functionals ω, q , setting $F := \text{EPS}_{\langle \rangle}^{\omega}(\varepsilon)(q)$ we have

$$\forall i \leq \omega F (\exists k < qF \varphi(i, k) \rightarrow \exists k < Fi \varphi(i, k)) \quad (17)$$

which is equivalent to the functional interpretation of $\Sigma_1\text{-CA}$. So what is the game-theoretic interpretation of our realizer F ? If we unravel (17) we see that we are essentially constructing a finite set

$$X_F := \{i \leq \omega F \mid \exists k < Fi \varphi(i, k)\}$$

that serves as an approximation to X with the property that if $i \leq \omega F$ has a witness for φ bounded by qF then we must have $i \in X_F$. In this sense ω and q can be read as set functions that calibrate the ‘size’ and ‘depth’ respectively of an approximation to X .

The set X_F is constructed as an optimal play in the game (ε, q, ω) . The job of the selection functions at round n is to decide whether or not to include the number n in the approximation, given that it has already made this decision for $\{0, \dots, n-1\}$. Its default is to omit n by playing 0, but if the resulting outcome $p_n(0)$ bounds some witness to n , it instead adds n and steals this witness as justification.

Therefore in this scenario the product of selection functions forms an intuitive set-theoretic construction, starting with the empty set and strategically

adding elements until it reaches the desired approximation. When interpreting a theorem that makes use of arithmetic comprehension as a lemma, we can simply plug in our realizer and impart its game theoretic meaning to better understand the realizer of the main theorem.

Some simple examples of well-known existence theorems that can be given a direct constructive interpretation using this instance of the product can be found in e.g. Simpson [21], such as the existence of maximal ideas in countable commutative rings or torsion subgroups in countable abelian groups. A more involved consequence of arithmetic comprehension using a more complex game, the Bolzano-Weierstrass theorem, will be discussed in the next section.

Example 16 (No injection $(\mathbb{N} \rightarrow \mathbb{N}) \rightarrow \mathbb{N}$). Following [18] we show that a higher type instance of the product that produces a sequence of *functions* can be used to effectively prove that there is no injection $\Psi: (\mathbb{N} \rightarrow \mathbb{N}) \rightarrow \mathbb{N}$ in any model of functionals in which the unbounded product exists. This time we consider the drinkers paradox applied to the formulas $P_n(f^{\mathbb{N} \rightarrow \mathbb{N}}) := (n = \Psi f)$. Defining

$$\varepsilon_n p := \begin{cases} p(\mathbf{0}) & \text{if } n = \Psi(p(\mathbf{0})) \\ \mathbf{0}^{\mathbb{N}} & \text{if } n \neq \Psi(p(\mathbf{0})), \end{cases}$$

where $\mathbf{0}^{\mathbb{N}} = \lambda k.0$, we have

$$\forall n, p(n = \Psi(p(\varepsilon_n P)) \rightarrow n = \Psi(\varepsilon_n p)).$$

As before, by Theorem 8, setting $F := \text{EPS}_{\emptyset}^{\omega}(\varepsilon)(q)$ we obtain

$$\forall n \leq \omega F(n = \Psi(qF) \rightarrow n = \Psi(F_n)). \quad (18)$$

Setting $qF \stackrel{\mathbb{N} \rightarrow \mathbb{N}}{:=} \lambda k.(F_k(k) + 1)$ we have a diagonal function that differs from each F_k at point k . Furthermore, if we set $\omega F = \Psi(qF)$, then from (18) on $i = \omega F = \Psi(qF)$ we obtain

$$\Psi(qF) = \Psi(F_{\Psi(qF)}).$$

But qF and $F_{\Psi(qF)}$ differ by definition, and we're done.

This is simply a constructive version of the usual diagonalisation argument used to prove that there is no injection from the real numbers to the natural numbers. The outcome functional $q\alpha$ is defined so that it differs from each of the αi on at least one value. As before, the ε_n plays a default value $\mathbf{0}$ and looks at the diagonal function $p_n(\mathbf{0})$ obtained by applying q to the optimal continuation of this move. If $\Psi(p_n(\mathbf{0}))$ it steals this witness and sets $\alpha n = p_n(\mathbf{0})$, else it is not concerned and sticks with the default move. The idea is to construct a sequence of representative functions α such that if Ψh for some h , then we must also have $\Psi(\alpha n)$. Of course we cannot effectively produce such an α , but using the product of selection functions we *can* produce an approximation that works at the point $\Psi(q\alpha)$, which is actually all that we need.

3.3 The product versus standard modes of recursion

A consequence of Theorem 14, and the fact that classical arithmetic and analysis can be formulated as classical logic plus finite and countable choice respectively,

is that the functional interpretation of classical proofs can be given entirely in terms of the product of selection functions. In fact we can reformulate Theorem 2 as follows, although we omit the details here and encourage the reader to consult [5, 8] instead.

Theorem 17. *We have*

- (a) PA^ω has a functional interpretation in primitive recursive arithmetic plus the finite product of selection functions (see [8] for details).
- (b) $\text{PA}^\omega + \text{AC}_0$ has a functional interpretation in primitive recursive arithmetic plus the unbounded product of selection functions.

It is natural then to ask how the product compares to those modes of recursion typically used in the functional interpretation of arithmetic and analysis.

Gödel's primitive recursive functionals of finite type [11] are the functional analogue of induction. By Theorem 14 (a) the finite product of selection functions is a functional analogue of finite choice, which is known to be equivalent to induction [20]. In [8] it is shown that the finite product is in fact equivalent to Gödel's primitive recursors over a weak λ -calculus, and thus offers an alternative construction of system T.

Countable choice and dependent choice are typically interpreted using Spector's bar recursion [22]. By Theorem 14 (b) and (c) we see that these are also interpreted by the unbounded product, and in [5] it is shown that bar recursion is primitive recursively equivalent to the unbounded product. The whole picture is sketched in Figure 1.

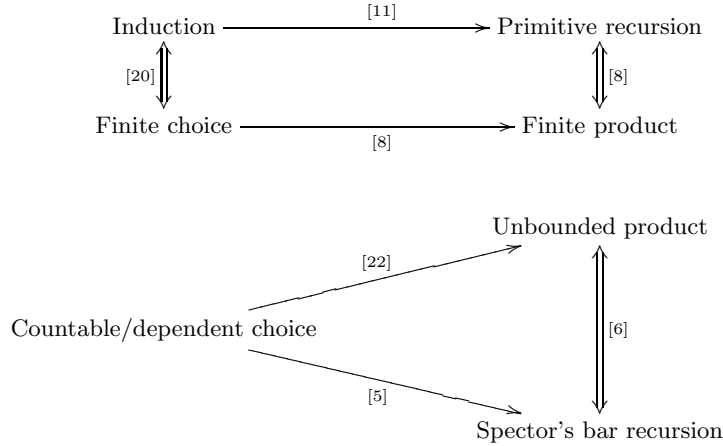


Figure 1: Functional interpretation of arithmetic and analysis

Of course our point here is that the advantage of using the product as opposed to the other modes of recursion is that it has a highly intuitive semantics, and witnesses extracted using the product often have a clear game-theoretic meaning. This is in stark contrast to other methods, particularly Spector's bar recursion, which are often very difficult to comprehend on a semantic level.

4 Interpreting the Bolzano-Weierstrass theorem

In this section we present a case study in which we formally extract a realizer for the functional interpretation of the Bolzano-Weierstrass theorem using the product of selection functions.

The constructive content of this theorem has been studied before, and in particular a detailed analysis using the dialectica interpretation and Spector's bar recursion is given in [15].

Our aim here is to show that, even though the Bolzano-Weierstrass theorem is relatively complex from a logical point of view, one can extract from its proof a program whose behaviour can be clearly understood, at least on an informal level, in terms of optimal strategies in sequential games.

As in [15] we analyse a proof of the theorem that combines countable choice with *weak König's lemma* - the statement that all infinite binary trees T have an infinite branch:

$$\text{WKL} : \forall n \exists s^{\mathbb{B}^n} T(s) \rightarrow \exists \alpha \forall n T([\alpha](n)).$$

We use the product of selection functions only to interpret the instance of countable choice used in the proof, as this forms the core of the extracted algorithm. In [12] Howard showed that WKL has a functional interpretation using only a weak, binary form of bar recursion. Rather than interpreting WKL using the product of selection functions (as done in [19]), for simplicity we just make use of Howard's realizer as its contribution to the complexity of the overall program is small compared to the main instance of the product.

4.1 The Bolzano-Weierstrass theorem.

The Bolzano-Weierstrass theorem states that any bounded sequence in \mathbb{R}^n has a convergent subsequence. Here we restrict ourselves to sequences of rationals (x_i) in the unit interval $[0, 1]$, as our analysis can be readily generalised. In the language of formal analysis, the Bolzano-Weierstrass theorem is given by

$$\text{BW}(x_i) : \exists a^{\mathbb{B}^{\mathbb{N}}}, b^{\mathbb{N}^{\mathbb{N}}} \forall n (bn < b(n+1) \wedge x_{bn} \in I_{[a](n)}),$$

where for a finite sequence of booleans s we define the interval

$$I_s := \left[\sum_{i=0}^{|s|-1} \frac{s_i}{2^{i+1}}, \sum_{i=0}^{|s|-1} \frac{s_i}{2^{i+1}} + \frac{1}{2^{|s|}} \right]$$

for $|s| > 0$ and $I_{\langle \rangle} := [0, 1]$. Intuitively a encodes a limit point $\bar{a} := \sum_{i=0}^{\infty} \frac{a_i}{2^{i+1}}$ of the sequence (x_i) and b defines a subsequence converging to this limit point, where $|x_{bn} - \bar{a}| \leq 2^{-(n-1)}$ for all n .

The functional interpretation of the Bolzano-Weierstrass theorem is given by

$$|\text{BW}(x_i)|_{\psi}^{A,B} = \forall n \leq \psi AB (Bn < B(n+1) \wedge x_{Bn} \in I_{[A](n)}),$$

where, to easy readability, we are omitting the dependency of A and B on ψ . The interpreted theorem states that there exist arbitrary large finite approximations B to a convergent subsequence, in the sense that $|x_{Bn} - \bar{A}| \leq 2^{-(n-1)}$ for all $n \leq \psi AB$.

4.2 A formal proof of $\text{BW}(x_i)$.

Assume an infinite sequence of rationals $(x_i)_{i \in \mathbb{N}}$ is fixed. Let us prove theorem $\text{BW}(x_i)$, i.e.

$$\exists a^{\mathbb{B}^{\mathbb{N}}}, b^{\mathbb{N}^{\mathbb{N}}} \forall n (bn < b(n+1) \wedge x_{bn} \in I_{[a](n)})$$

directly using WKL and AC_0 . We define the predicate T by

$$T(s^{\mathbb{B}^*}, k) := |s| < k \wedge \exists i \in (|s|, k] (x_i \in I_s). \quad (19)$$

We write \mathbb{B}^* for finite sequences of booleans, and \mathbb{B}^n for sequences of booleans of length n .

Lemma 18. *By countable choice AC_0 there exists a function $\beta: \mathbb{N} \rightarrow \mathbb{N}$ such that*

$$\forall n, s^{\mathbb{B}^n} (\exists k T(s, k) \rightarrow T(s, \beta n)). \quad (20)$$

Proof. By the drinkers paradox we have

$$\forall n, s^{\mathbb{B}^n} \exists l (\exists k T(s, k) \rightarrow T(s, l)).$$

By bounded collection and the fact that T has the monotonicity property $T(s, k) \rightarrow T(s, k+l)$ we have

$$\forall n \exists L \forall s^{\mathbb{B}^n} (\exists k T(s, k) \rightarrow T(s, L)). \quad (21)$$

Finally, by countable choice we obtain β satisfying (20). \square

For the rest of the section let β be a function whose existence is shown in the above lemma, so β satisfies (20). Also, define $T^\beta(s) := T(s, \beta(|s|))$, so that $\exists k T(s, k) \leftrightarrow T^\beta(s)$.

Corollary 19. *$T^\beta(s)$ is a decidable tree predicate.*

Proof. $T^\beta(s)$ is clearly decidable in the given oracle β . It remains to see that it is prefix-closed. Observe that $T(s * t, \beta(|s * t|)) \rightarrow T(s, \beta(|s * t|))$, by the definition of T . Also, by (20), $T(s, \beta(|s * t|)) \rightarrow T(s, \beta(|s|))$. Combining the two we have $T^\beta(s * t) \rightarrow T^\beta(s)$. \square

Hence, given a decidable binary tree predicate, we can apply weak König's lemma to obtain the following:

Lemma 20. *There exists a sequence $a: \mathbb{B}^{\mathbb{N}}$ such that*

$$\forall n \underbrace{(n < \beta n \wedge \exists i \in (n, \beta n] (x_i \in I_{[a](n)}))}_{T^\beta([a](n))}. \quad (22)$$

Proof. This follows from WKL applied to T^β , once we have shown that the tree $T^\beta(s)$ has branches of arbitrary length. To see that, fix n and let s , with $|s| = n$, be the index of the interval I_s which contains x_n . Then, clearly we have

$$|s| < n+1 \wedge x_n \in I_s$$

which implies $T(s, n+1)$. By (20) we obtain $T(s, \beta(|s|))$. For future reference, we call h this function producing s for each given n . \square

Theorem 21 (Bolzano-Weierstrass). *Given a sequence of rationals $(x_i)_{i \in \mathbb{N}}$, there exists $a: \mathbb{B}^{\mathbb{N}}$ and $b: \mathbb{N}^{\mathbb{N}}$ such that*

$$\forall n (bn < b(n+1) \wedge x_{bn} \in I_{[a](n)}). \quad (23)$$

Proof. Let a be as in Lemma 20. Define b by

$$b0 := 0 \quad (24)$$

$$b(n+1) := \mu i \in (bn+1, \beta(bn+1)] (x_i \in I_{[a](bn+1)}).$$

By (22) such least $i \in (bn+1, \beta(bn+1)]$ satisfying $x_i \in I_{[a](bn+1)}$ always exists. Clearly we have $bn < b(n+1)$, and also $x_{b(n+1)} \in I_{[a](bn+1)} \subseteq I_{[a](n+1)}$, since $bn+1 \geq n+1$. \square

4.3 Lemma 18 via the product of selection functions

We first interpret our main instance of countable choice (Lemma 18). We want to produce an approximation to the function β given counterexample functions $\omega, q: \mathbb{N}^{\mathbb{N}} \rightarrow \mathbb{N}$ for n and k respectively in (20):

$$\forall \omega, q \exists \beta \forall n \leq \omega \beta \forall s^{\mathbb{B}^n} (T(s, q\beta) \rightarrow T(s, \beta n)). \quad (25)$$

First we need to find selection functions $\delta_n: \mathbb{N}^{\mathbb{N}} \rightarrow \mathbb{N}$ witnessing the functional interpretation of (21):

$$\exists \delta \forall n, p \forall s^{\mathbb{B}^n} (T(s, p(\delta_n p)) \rightarrow T(s, \delta_n p)). \quad (26)$$

Since (21) is just the drinkers paradox combined with bounded collection, appropriate selection functions are constructed in a similar manner to (10).

Lemma 22. *Let δ_n be defined as*

$$\delta_n p := p^i(0) \quad (27)$$

where i is the least $\leq 2^n$ such that, for all $s^{\mathbb{B}^n}$, $T(s, p^{i+1}(0)) \rightarrow T(s, p^i(0))$. Then δ witnesses (26).

Proof. Note that (26) holds by definition once we show that such $i \leq 2^n$ must exist. Assume, for the sake of a contradiction, that

(I) for all $i \leq 2^n$ there exists an $s^{\mathbb{B}^n}$ such that $T(s, p^{i+1}(0))$ and $\neg T(s, p^i(0))$.

Because $T(s, k)$ is monotone on the second argument, (I) implies that

(II) $p^i(0) < p(p^i(0))$, for all $0 \leq i \leq 2^n$.

Since, in (I), we have $2^n + 1$ possible values for i but only 2^n possible values for s , there must be an s and distinct i and j , say $i < i+1 \leq j$, such that $T(s, p^{i+1}(0))$ and $\neg T(s, p^j(0))$. By (II), however, that is a contradiction. \square

Theorem 23. *Given $q: \mathbb{N}^{\mathbb{N}} \rightarrow \mathbb{N}$ and $\omega: \mathbb{N}^{\mathbb{N}} \rightarrow \mathbb{N}$ let $\beta: \mathbb{N}^{\mathbb{N}}$ be defined as*

$$\beta := \text{EPS}_{\omega}^{\omega}(\delta)(q),$$

with δ as in (27). Then β witnesses (25).

Proof. By Lemma 22, the δ_n as defined in (27) are such that

$$\forall n, p \forall s^{\mathbb{B}^n} (T(s, p(\delta_n p)) \rightarrow T(s, \delta_n p)).$$

For $n \leq \omega \beta$ and $p = p_{[\beta](n)}$, by Theorem 8 we have $\beta n = \delta_n p$ and $q\beta = p(\delta_n p)$, from which we can conclude $\forall s^{\mathbb{B}^n} (T(s, q\beta) \rightarrow T(s, \beta n))$. \square

4.4 A realizer for $\text{BW}(x_i)$

Finally, we show how the instance of the product of selection functions in Theorem 23, used to interpret the crucial Lemma 18, lies behind an algorithm for constructing approximations to BW . We first interpret Lemma 20, making use of Howard's realizer for WKL using a weak, binary form of bar recursion, full details of which can be found in [12].

Lemma 24. *For any counterexample function $\varphi: \mathbb{B}^{\mathbb{N}} \times \mathbb{N}^{\mathbb{N}} \rightarrow \mathbb{N}$ there exist $\beta: \mathbb{N}^{\mathbb{N}}$ and $A: \mathbb{B}^{\mathbb{N}}$ satisfying*

$$\forall n \leq \varphi A \beta \underbrace{(n < \beta n \wedge \exists i \in (n, \beta n] (x_i \in I_{[A](n)}))}_{T([A](n), \beta n)}. \quad (28)$$

Proof. Assume $\varphi: \mathbb{B}^{\mathbb{N}} \times \mathbb{N}^{\mathbb{N}} \rightarrow \mathbb{N}$ given. For any given $\beta: \mathbb{N}^{\mathbb{N}}$ let $N^\beta: \mathbb{B}^* \rightarrow \mathbb{N}$ be defined via Howard's binary bar recursion as

$$N^\beta(t) := \begin{cases} 0 & \text{if } \exists s \preceq t (\varphi(\hat{s}, \beta) < |s|) \\ 1 + \max\{N^\beta(t * 0), N^\beta(t * 1)\} & \text{otherwise.} \end{cases}$$

We first construct β as in Theorem 23 where we set

$$q\beta = \omega\beta := N^\beta(\langle \rangle) + 1,$$

to obtain

$$\forall n \leq N^\beta(\langle \rangle) \forall s \in \mathbb{B}^n (T(s, N^\beta(\langle \rangle) + 1) \rightarrow T(s, \beta n)). \quad (29)$$

Let $N = N^\beta$ for β as just defined. It can be shown (cf. Howard [13]) that for any $|t| \geq N(\langle \rangle)$ there is some $s \preceq t$ with $\varphi(\hat{s}, \beta) < |s|$. Therefore we define

$$A := \hat{t}, \quad \text{where } t = \mu s \preceq h(N(\langle \rangle)) (\varphi(\hat{s}, \beta) < |s|)$$

where h is defined as in the proof of Lemma 20. Now, by the definition of h we have $T(h(N(\langle \rangle)), N(\langle \rangle) + 1)$. Also, for $n \leq \varphi(A, \beta)$, by the definition of t (in definition of A) we have $n < |t|$, and hence $[A](n) \preceq h(N(\langle \rangle))$. Therefore, by the definition of $T(s, k)$ we also have $T([A](n), N(\langle \rangle) + 1)$. Finally, by (29) we get $T([A](n), \beta n)$, and so we have proved (28). \square

We are in a position now to effectively witness an approximation to the real Bolzano-Weierstrass (Theorem 21).

Theorem 25. *For any counterexample function $\psi: \mathbb{B}^{\mathbb{N}} \times \mathbb{N}^{\mathbb{N}} \rightarrow \mathbb{N}$ there exists $A: \mathbb{B}^{\mathbb{N}}$ and $B: \mathbb{N}^{\mathbb{N}}$ satisfying*

$$\underbrace{\forall n \leq \psi AB (Bn < B(n+1) \wedge x_{Bn} \in I_{[A](n)})}_{|\text{BW}(x_i)|_\psi^{A,B}}.$$

Proof. Let $b_{a,\beta}$ denote the construction in (24), and let

$$\tilde{\beta}(n) := \max_{i \leq n+1} \beta(i).$$

Define

$$\varphi(A, \beta) := \tilde{\beta}^{\psi(A, b_{A,\beta})}(0).$$

Construct A and β as in Lemma 24 using counterexample function φ as above defined, and let $B := b_{A,\beta}$. By (28) we have

$$\forall n \leq \tilde{\beta}^{\psi AB}(0) \left(n < \beta n \wedge \exists i \in (n, \beta n] \left(x_i \in I_{[A](n)} \right) \right). \quad (30)$$

This is an approximation of Lemma 20, but it is enough to obtain our desired approximation of Theorem 21. Indeed, for all $n \leq \psi AB$ we are guaranteed to have, by (30), that $\beta n < B(n+1)$ and $x_{\beta n} \in I_{[A](n)}$. \square

4.5 Understanding the realizer for $\text{BW}(x_i)$

Let us first take an informal look at the game that forms the basis of the interpretation of countable choice given in Section 4.3. The strategy at round n is to pick a number m satisfying

$$\exists j \in (n, p(m)] \left(x_j \in I_s \right) \rightarrow \exists j \in (n, m] \left(x_j \in I_s \right)$$

for all $s: \mathbb{B}^n$. In other words, given a local outcome function $p: \mathbb{N} \rightarrow \mathbb{N}$, the selection function picks a number $m = \delta_n p$ such that whenever the interval I_s contains some x_j where j is bounded by the outcome $p(m)$ of playing m , then I_s also contains some x_j with j bounded by m .

The selection function δ_n prescribes what is essentially an iterated version of the strategy given for the drinkers paradox (10). It first attempts to play $m = 0$ provided that $\forall s^{\mathbb{B}^n} \forall j \in (n, p(0)] \left(x_j \notin I_s \right)$. But this cannot be the case unless $p(0) \leq n$. Hence, if $p(0) > n$, there exists an s and a $j \in (n, p(0)]$ with $x_j \in I_s$. It then continues and attempts to play $p(0)$ in the hope that no additional intervals I_s contain x_j for $j \in (p(0), p^2(0)]$. Continuing along these lines, it is not difficult to see that $m = p_n^k(0)$ must work for some $k \leq 2^n$ since there are only 2^n intervals I_s , with $|s|$.

The resulting optimal strategy in the game (δ, ω, q) is a sequence β that acts as an approximation to the function ineffectively constructed in Lemma 18:

$$\forall n \leq \omega \beta \forall s^{\mathbb{B}^n} \left(\exists k \in (n, q\beta] \left(x_k \in I_s \right) \rightarrow \exists j \in (n, \beta n] \left(x_j \in I_s \right) \right).$$

The power of this procedure is evident when we observe that by constructing the outcome function q and control function ω as in Section 4.4 (incorporating Howard's realizer for WKL), the resulting optimal play β can be directly used to construct approximations A and B for $\text{BW}(x_i)$.

Our aim here has been to convince the reader that while constructing a realizer for the functional interpretation of the Bolzano-Weierstrass theorem takes a reasonable amount of work, the game theoretic intuition behind the product of selection functions allows us to gain a better understanding of the key operational features of this realizer.

5 Further remarks

In this article we have shown that the language of selection functions and sequential games underlies the dialectica interpretation of classical proofs in a fundamental way, and we have used the product of selection functions to construct a concise and intuitive computational interpretation of some well-known theorems.

Our motivation has been a more qualitative understanding of functional interpretations, as a response the fact that formal proof-theoretic methods are becoming increasingly relevant in modern mathematics. We have shown that the product of selection functions is a fundamental construction behind the dialectica interpretation of classical proofs, and we hope to have convinced the reader that in practise it leads to extracted programs that have an expressive reading in terms of optimal strategies in sequential games.

There is a lot of work to be done towards understanding formal proof-theoretic techniques in mathematical terms, and the question of adapting and refining functional interpretations so that they can be seen as intelligent translations on mathematical proofs as opposed to just syntactic translations on logic sentences forms a very interesting area of research. The authors believe that there are several potentially fruitful avenues for further research.

One is to explore in more detail the link between the dialectica interpretation and the closely related ‘correspondence principle’ implicitly used in areas like ergodic theory. In particular, the finitary version of theorems discussed by Tao in [23, 24] are strictly speaking related to the *monotone* variant of the dialectica interpretation, which extracts uniform bounds, or majorants, for realizers of the interpretation. It would be interesting to try to gain an understanding of how the product of selection functions can be used to extract realizers for the monotone interpretation and therefore produce constructive proofs of theorems that can truly be seen as ‘finitizations’ in the sense of Tao.

Another interesting issue is the *efficiency* of the product of selection functions in producing a realizer. For instance, a quick analysis of Example 16 shows that if $\Psi(\lambda n.0) = \Psi(\lambda n.1)$, our program potentially misses this obvious counterexample and eventually produces a much more elaborate one. This highlights the fact that while the product is indeed an intuitive realizer for the axiom of choice, it is far from optimal and refinements of the procedure or even completely different recursion schemata may be more suited to interpreting specific principles. An example of this is open recursion proposed by Berger in [4] for the realizability interpretation of the minimal bad sequence argument.

A related question is the efficiency of the dialectica interpretation itself, and the comparison of extracted programs to those obtained using other proof interpretations such as modified realizability. In particular, the modified realizability interpretation of choice has an interesting realizer given by Berardi et al. [3] that was also shown to have a natural game theoretic reading.

We conclude with the remark that our paper belongs to a larger body of recent work on the theory of selection functions and sequential games by the first author and M. Escardó, starting with [6] and surveyed in [7]. Particularly relevant is [19], in which the product of selection functions is used to extract a game theoretic realizer for Ramsey’s theorem. This can be seen as an extended case study illustrating the methods employed here.

References

- [1] J. Avigad and S. Feferman. Gödel’s functional (“Dialectica”) interpretation. In S. R. Buss, editor, *Handbook of proof theory*, volume 137 of *Studies in Logic and the Foundations of Mathematics*, pages 337–405. North Holland, Amsterdam, 1998.

- [2] J. Avigad, P. Gerhardy, and H. Towsner. Local stability of ergodic averages. *Transactions of the American Mathematical Society*, 362:261–288, 2010.
- [3] S. Berardi, M. Bezem, and T. Coquand. On the computational content of the axiom of choice. *The Journal of Symbolic Logic*, 63(2):600–622, 1998.
- [4] Ulrich Berger. A computational interpretation of open induction. In F. Titsworth, editor, *Proceedings of the Ninetenth Annual IEEE Symposium on Logic in Computer Science*, pages 326–334, 2004.
- [5] M. H. Escardó and P. Oliva. Computational interpretations of analysis via products of selection functions. In F. Ferreira, B. Lowe, E. Mayordomo, and L. M. Gomes, editors, *Computability in Europe 2010, LNCS 6158*, pages 141–150. Springer, 2010.
- [6] M. H. Escardó and P. Oliva. Selection functions, bar recursion, and backward induction. *Mathematical Structures in Computer Science*, 20(2):127–168, 2010.
- [7] M. H. Escardó and P. Oliva. Sequential games and optimal strategies. *Royal Society Proceedings A*, 467:1519–1545, 2011.
- [8] M. H. Escardó, P. Oliva, and T. Powell. System T and the product of selection functions. In Marc Bezem, editor, *Computer Science Logic (CSL’11)*, volume 12 of *LIPICs*, pages 233–247, 2011.
- [9] H. Friedman. Classically and intuitionistically provably recursive functions. In D. Scott and G. Müller, editors, *Higher Set Theory*, volume 669 of *Lecture Notes in Mathematics*, pages 21–28. Springer, Berlin, 1978.
- [10] J. Gaspar and U. Kohlenbach. On Tao’s “finitary” infinite pigeonhole principle. *Journal of Symbolic Logic*, 75(1):355–371, 2010.
- [11] K. Gödel. Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes. *Dialectica*, 12:280–287, 1958.
- [12] W. A. Howard. Ordinal analysis of bar recursion of type zero. *Compositio Mathematica*, 42:105–119, 1981.
- [13] W. A. Howard. Ordinal analysis of simple cases of bar recursion. *The Journal of Symbolic Logic*, 46:17–30, 1981.
- [14] U. Kohlenbach. *Applied Proof Theory: Proof Interpretations and their Use in Mathematics*. Monographs in Mathematics. Springer, 2008.
- [15] U. Kohlenbach and P. Safarik. On the computational content of the Bolzano-Weierstrass principle. *Math. Logic Quartely*, 56:508–532, 2010.
- [16] G. Kreisel. On the interpretation of non-finitist proofs, part I. *The Journal of Symbolic Logic*, 16:241–267, 1951.
- [17] G. Kreisel. On the interpretation of non-finitist proofs, part II: Interpretation of number theory. *The Journal of Symbolic Logic*, 17:43–58, 1952.

- [18] P. Oliva. Understanding and using Spector’s bar recursive interpretation of classical analysis. In A. Beckmann, U. Berger, B. Löwe, and J. V. Tucker, editors, *Proceedings of CiE’2006, LNCS 3988*, pages 423–234. Springer, 2006.
- [19] P. Oliva and T. Powell. A constructive interpretation of Ramsey’s theorem via the product of selection functions. Preprint, 2011.
- [20] C. Parsons. On a number theoretic choice schema and its relation to induction. In A. Kino, J. Myhill, and R. E. Vesley, editors, *Intuitionism and Proof Theory: Proceedings of the Summer Conference at Buffalo, N.Y. 1968*, pages 459–473. North Holland, Amsterdam, 1970.
- [21] S. G. Simpson. *Subsystems of Second Order Arithmetic*. Perspectives in Mathematical Logic. Springer, Berlin, 1999.
- [22] C. Spector. Provably recursive functionals of analysis: a consistency proof of analysis by an extension of principles in current intuitionistic mathematics. In F. D. E. Dekker, editor, *Recursive Function Theory: Proc. Symposia in Pure Mathematics*, volume 5, pages 1–27. American Mathematical Society, Providence, Rhode Island, 1962.
- [23] T. Tao. Soft analysis, hard analysis, and the finite convergence principle. Essay, available at <http://terrytao.wordpress.com/2007/05/23/soft-analysis-hard-analysis-and-the-finite-convergence-principle/>, 2007.
- [24] T. Tao. The correspondence principle and finitary ergodic theory. Essay, available at <http://terrytao.wordpress.com/2008/08/30/the-correspondence-principle-and-finitary-ergodic-theory/>, 2008.